Remarks

This Application has been carefully reviewed in light of the Office Action mailed April 23, 2007. Applicant appreciates the Examiner's consideration of the Application. Applicant believes all claims are allowable without amendment and respectfully provides the following remarks. Applicant respectfully requests reconsideration and allowance of all pending claims.

I. The Claims are Allowable over *Alumbaugh*

The Examiner rejects Claims 1-30 under 35 U.S.C. § 102(a) as being anticipated by U.S. Publication No. 2003/0172368 A1 filed by Alumbaugh et al. ("*Alumbaugh*"). Applicant respectfully disagrees.

"A claim is anticipated only if ***each and every element as set forth in the claim*** is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 U.S.P.Q.2d 1051, 1053 (Fed. Cir. 1987) (emphasis added); M.P.E.P. ch. 2131. In addition, "[t]he identical invention must be shown in ***as complete detail as contained in the . . . claim***." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989) (emphasis added); *see also* M.P.E.P. ch. 2131. Furthermore, "[t]he elements must be arranged ***as in the claim*** under review." *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990) (emphasis added); M.P.E.P. ch. 2131. As illustrated below, the cited portions of *Alumbaugh* do not appear to disclose, either expressly or inherently, each and every limitation recited in Applicant's independent claims.

A. Independent Claims 1 and 8 and their Dependent Claims are Allowable

The cited portions of *Alumbaugh* do not appear to disclose, teach, or suggest at least the following limitations recited in Claim 1 (which Applicant discusses as an example):

- pars[ing] a plurality of transaction definitions for a software system, wherein each transaction definition comprises one or more parameters;
- generat[ing] a plurality of schema definitions in response to the plurality of transaction definitions, wherein the schema definitions are written in a self-describing language;

- wherein a first schema definition is operable to map the one or more parameters associated with a first transaction definition to a first document written in the self-describing language; and
- wherein a second schema definition is operable to map a second document written in the self-describing language to the one or more parameters associated with a second transaction definition.

For example, the cited portion of *Alumbaugh* does not appear to disclose, teach, or suggest "pars[ing] a plurality of transaction definitions for a software system, wherein each transaction definition comprises one or more parameters," as recited in Claim 1. As allegedly disclosing these limitations, the Examiner cites Page 2, Paragraph 0095 of *Alumbaugh.* (Office Action, Page 3) The cited portion of *Alumbaugh* defines a "Validating XML Parser" as "a parser that, when parsing XML, validates both that the XML is well formed and that the XML is valid based on the rules in a specified DTD or XML-Schema file." (Page 2, Paragraph 0095) While the cited portion of *Alumbaugh* mentions parsing, the cited portion does not appear to disclose, teach, or suggest "pars[ing] ***a plurality of transaction definitions for a software system***, wherein ***each transaction definition comprises one or more parameters***," as recited in Claim 1. It does not appear, for example, that the cited portion of *Alumbaugh* makes any mention of "a plurality of transaction definitions for a software system," let alone parsing a plurality of transaction definitions for a software system or that each transaction definition comprises one or more parameters.

As another example, at least because the cited portions of *Alumbaugh* do not disclose, teach, or suggest "pars[ing] a plurality of transaction definitions for a software system, wherein each transaction definition comprises one or more parameters," as recited in Claim 1, the cited portions of *Alumnbaugh* necessarily fail to disclose, teach, or suggest "generat[ing] a plurality of schema definitions ***in response to the plurality of transaction definitions***, wherein the schema definitions are written in a self-describing language," as recited in Claim 1.

As another example, at least because the cited portions of *Alumbaugh* do not disclose, teach, or suggest "generat[ing] a plurality of schema definitions in response to the plurality of transaction definitions, wherein the schema definitions are written in a self-describing language," as recited in Claim 1, the cited portions of *Alumbaugh* necessarily fail to disclose, teach, or suggest "wherein a first schema definition is operable to map the one or more parameters associated with a first transaction definition to a first document written in the self-describing language" and "wherein a second schema definition is operable to map a second document written in the self-describing language to the one or more parameters associated with a second transaction definition," as recited in Claim 1. As allegedly disclosing these limitations recited in Claim 1, the Examiner cites Page 10, Paragraph 0229 (particularly, the similarity maps) in *Alumbaugh*. (Office Action, Page 3) The cited portion of *Alumbaugh* states the following:

> Once the Assessment Micro Agent has been installed in two or more applications, it is possible to produce similarity maps between those applications based on the data structure inventory provided by it. In order to accomplish this, the Application Ontology Factory uses application data structure information provided by the Assessment Micro Agent and the information provided in the Common Ontology library to produce the application ontologies. Then the App2App Similarity Mapper then uses the information in the application ontologies to produce a similarity map between the applications. Once the similarity map is completed, the Planner uses the information contained in the similarity map to produce an integration plan. Then the CodeGen Agent uses the information provided in the integration plan to produce the integration code. After the integration code is validated by the Error Management Micro Agent, the it is deployed as the x-walk file between the applications and thus they become integrated.

(Page 10, Paragraph 0229)

Thus, the cited portion of *Alumbaugh* discloses the ability to produce similarity maps between applications in which the Assessment Micro Agent is installed. However, the cited portion of *Alumbaugh* does not appear to disclose, teach, or suggest the particular mapping recited in Claim 1. That is, the cited portion of *Alumbaugh* does not appear to disclose, teach, or suggest "a first schema definition [that] is operable to map the one or more parameters associated with a first transaction definition to a first document written in the self-describing language" and "a second schema definition [that] is operable to map a second document written in the self-describing language to the one or more parameters associated with a second transaction definition," as recited in Claim 1.

For at least these reasons, Applicant respectfully requests reconsideration and allowance of independent Claim 1 and its dependent claims. For at least certain analogous reasons, Applicant respectfully requests reconsideration and allowance of independent Claim 8 and its dependent claims.

B. Independent Claims 12 and 22 and their Dependent Claims are Allowable

The cited portions of *Alumbaugh* fail to disclose, teach, or suggest at least the following limitations recited in Claim 12 (which Applicant discusses as an example):

- a software service operable to receive a transaction request and to generate a first object associated with the transaction request;
- an object generator operable to convert the first object into a first document written in a self-describing language; and
- a document generator operable to convert the first document into a first transaction message according to a schema associated with a first transaction type determinable from the first document.

For example, the cited portions of *Alumbaugh* do not appear to disclose, teach, or suggest "a software service operable to receive a transaction request and to generate a first object associated with the transaction request," as recited in Claim 12. As allegedly disclosing these limitations, the Examiner cites the "Assessment Micro Agent" disclosed as item 320 in *Alumbaugh.* (Office Action, Page 6) *Alumbaugh* appears to discuss the Assessment Micro Agent at Page 10, Paragraph 0231 through Page 11, Paragraph 0241. This portion of *Alumbaugh* discloses that the Assessment Micro Agent serves three primary functions: schema discover, change monitoring, and system or user notification of changes. (Page 10, Paragraph 0232) According to *Alumbaugh*, schema discover involves reading the meta-data stored in a data source 310 to produce a schema 360 that is placed into a memory model, which can then be displayed in textural 380 or graphic 390 form. (Page 10, Paragraph 0233) *Alumbaugh* discloses that change monitoring provides detailed analysis through the Change Specification Manager 8 of software under consideration so that the user knows exactly what is different between product versions. (Page 10, Paragraph 0234) The change specification can be show to a user. (*Id.*) However, the cited portion of *Alumbaugh* does not appear to disclose, teach, or suggest that the Assessment Micro Agent is "operable to ***receive a transaction request*** and to ***generate a first object associated with the transaction request***," as recited in Claim 12.

As another example, at least because the cited portions of *Alumbaugh* do not disclose, teach, or suggest "a software service operable to receive a transaction request and to generate a first object associated with the transaction request," as recited in Claim 12, the cited portions of *Alumbaugh* necessarily fail to disclose, teach, or suggest "an object generator operable to convert the first object into a first document written in a self-describing language," as recited in Claim 12. As allegedly disclosing the object generator recited in Claim 12, the Examiner cites the "Schema Manager" disclosed as item 4 in *Alumbaugh*. (Office Action, Page 6) *Alumbaugh* appears to define the Schema Manager at Page 11, Paragraph 0237. According to *Alumbaugh*, the Schema Manager connects to applications through standard interfaces (e.g., JDBC, ODBC, Flat File Translators). (Page 11, Paragraph 0237) The Schema Manager makes an analysis of the application and extracts the meta-data model in the form of a schema. (*Id.*) The Schema Manager stores the schema and then provides an interface to other components to retrieve the schema when necessary. (*Id.*) For instance, the Change Specification Manager retrieves schemas to produce change specifications on the schemas. (*Id.*) The schema manager also allows the schemas to be exported into other formats, including XML, Serialized Java Objects, HTML, and others. (*Id.*) However, the cited portion of *Alumbaugh* does not appear to disclose, teach, or suggest that the Schema Manager (which the Examiner equates with the object generator recited in Claim 12) is "operable to convert the first object into a first document written in a self-describing language," as recited in Claim 12.

As another example, at least because the cited portions of *Alumbaugh* does not disclose, teach, or suggest "a software service operable to receive a transaction request and to generate a first object associated with the transaction request" and "an object generator operable to convert the first object into a first document written in a self-describing language," as recited in Claim 12, the cited portions of *Alumbaugh* necessarily fail to disclose, teach, or suggest "a document generator operable to convert the first document into a first transaction message according to a schema associated with a first transaction type determinable from the first document," as recited in Claim 12.

For at least these reasons, Applicant respectfully requests reconsideration and allowance of independent Claim 12 and its dependent claims. For at least certain analogous reasons, Applicant respectfully requests reconsideration and allowance of independent Claim 22 and its dependent claims.

II. No Waiver

All of Applicant's arguments and amendments are without prejudice or disclaimer. Additionally, Applicant has merely discussed example distinctions from the reference cited by the Examiner. Other distinctions may exist, and Applicant reserves the right to discuss these additional distinctions in a later Response or on Appeal, if appropriate. By not responding to additional statements made by the Examiner, Applicant does not acquiesce to the Examiner's additional statements. The example distinctions discussed by Applicant are sufficient to overcome the Examiner's rejections.

Conclusion

Applicant has made an earnest attempt to place the Application in condition for allowance. For at least the foregoing reasons, Applicant respectfully requests full allowance of all pending claims.

If the Examiner feels that a telephone conference or an interview would advance prosecution of the Application in any way, the Examiner is invited to contact the undersigned attorney for Applicant at the Examiner's convenience at (214) 953-6813.

Although Applicant believes no fees are due, the Commissioner is hereby authorized to charge any necessary fees or credit any overpayment to Deposit Account No. 05-0765 of Electronic Data Systems Corporation.

Respectfully submitted,

BAKER BOTTS L.L.P.
Attorneys for Applicant

Chad D. Terrell
Reg. No. 52,279

Date: July 23, 2007

Customer Number: **35005**